

# **EXMARaLDA – ein System zur computergestützten Diskurstranskription**

*Thomas Schmidt*

## **1 Einleitung**

Dieser Aufsatz beschreibt EXMARaLDA, ein XML-basiertes System zur computergestützten Diskurstranskription, das seit zwei Jahren am Sonderforschungsbereich „Mehrsprachigkeit“ an der Universität Hamburg entwickelt wird. Im ersten Abschnitt wird zunächst die Rolle dieses Systems als zentrale Architekturkomponente einer Datenbank „Mehrsprachigkeit“ erläutert. Der zweite Abschnitt legt dann die konzeptionellen Details der XML-Formate dar und gibt einen Überblick über die zum System gehörigen Ein- und Ausgabemethoden. Schließlich wird im dritten Abschnitt versucht, einige grundlegende Probleme anzureißen, die sich beim Einsatz von XML zur Kodierung von Diskurstranskriptionen ergeben.

## **2 Datenbank „Mehrsprachigkeit“**

Der Sonderforschungsbereich 538 „Mehrsprachigkeit“ vereinigt in dreizehn Teilprojekten eine Vielzahl von Forschern, die sich unter verschiedensten theoretischen Perspektiven mit dem Thema der Mehrsprachigkeit auseinandersetzen. Die weitaus größte Zahl der Projekte arbeitet dabei empirisch auf der Grundlage von Aufnahmen gesprochener Sprache, die durch eine computergestützte Transkription der Analyse zugänglich gemacht werden. Wie dies im Einzelnen geschieht, variiert stark von Projekt zu Projekt. Unterschiede finden sich zunächst auf der konzeptionellen Ebene. Je nach Projektzusammenhang, d.h. je nach theoretischem Hintergrund, vorrangigem Untersuchungsziel und untersuchten Einzelsprachen werden unterschiedliche Transkriptionskonventionen benutzt, die ihrerseits verschiedene Diskursmodelle widerspiegeln. Dies schlägt sich zum Beispiel in der Auswahl der zu transkribierenden Phänomenbereiche wieder, aber auch in den graphischen Organisationsformen der Transkripte. Zwei gegensätzliche Beispiele mögen dies verdeutlichen:

Das Projekt B2 „Simultaner und sukzessiver Erwerb von Mehrsprachigkeit“ untersucht Syntaxerwerb bei mehrsprachig (französisch/deutsch, portugiesisch/deutsch oder baskisch/spanisch) aufwachsenden Kindern vor dem Hintergrund der generativen Grammatik. Die Transkripte werden in Spaltennotation angefertigt, da diese graphische Organisationsform als besonders geeignet angesehen wird, asymmetrisch organisierte, d.h. in diesem Falle kindzentrierte, Dis-

kursformen darzustellen (vgl. Edwards 1992). Da die Analysen vornehmlich auf die Konstituentenstruktur einzelner Sätze zielen, werden Performanzphänomene gesprochener Sprache (gefüllte Pausen etc.) und zeitlichen Abfolgen von Diskursereignissen (Überlappung von Äußerungen, redebegleitende Gesten etc.) bei der Transkription relativ wenig Beachtung geschenkt.

Im Gegensatz dazu sind im Projekt K1 „Japanische und deutsche Expertendiskurse in ein- und mehrsprachigen Konstellationen“ gerade solche Phänomene für die Analyse von ausschlaggebender Bedeutung. Das Projekt untersucht Handlungsmuster und Diskursarten vor dem Hintergrund der Funktionalen Pragmatik. Die Transkriptionen werden nach dem Verfahren der Halbinterpretativen Arbeitstranskription (HIAT, Ehlich/Rehbein 1976) angefertigt. Das Verfahren arbeitet mit der Partiturnotation, weil diese graphische Organisationsform gut geeignet ist, Simultaneität und Sukzessivität verschiedener gleichberechtigter Handlungsstränge übersichtlich darzustellen (vgl. Edwards 1992). Da die Untersuchungen primär diskursanalytisch orientiert sind, finden Performanzphänomene und non-verbale Kommunikation bei der Transkription detaillierte Beachtung.

Die durch den Analysezweck motivierte Wahl bestimmter Transkriptionsverfahren und Darstellungsweisen für Transkripte schlägt sich weiterhin in der Wahl der Software-Werkzeuge nieder, mittels derer sie umgesetzt werden. Das Anfertigen von Transkripten in Partiturnotation setzt spezialisierte Software voraus, die Ein- und Ausgabe von interlinearem Text unterstützt. Mehrere Projekte<sup>1</sup> arbeiten deshalb mit syncWriter (Rehbein *et al.* 1993), ein Projekt benutzt HIAT-DOS<sup>2</sup> (Ehlich 1992). Die mit Spaltennotation arbeitenden Projekte<sup>3</sup> fertigen die Transkripte zunächst handschriftlich an und erstellen dann auf dieser Grundlage Satzdatenbanken (LAPSUS<sup>4</sup>, Crysman 1995), die Annotation und Auswertung der Daten nach syntaktischen Gesichtspunkten vereinfachen, dabei aber nicht die ganze strukturelle Komplexität des Transkriptes übernehmen. Schließlich benutzen ein Projekt<sup>5</sup> herkömmliche Texteditoren, um Transkripte in vertikaler Notation, beispielsweise nach den Verbmobil-Konven-

- 1 Projekte K1 „Japanische und Deutsche Expertendiskurse“, K2 „Dolmetschen im Krankenhaus“ und E5 „Sprachliche Konnektivität bei bilingual türkisch-deutsch aufwachsenden Kindern“.
- 2 Projekt K5 „Semikommunikation und rezeptive Mehrsprachigkeit im heutigen Skandinavien“
- 3 Projekte E1 „Frühkindliche Zweisprachigkeit: Italienisch/Deutsch und Französisch/Deutsch im Vergleich“, E2 „Simultaner und Sukzessiver Erwerb von Mehrsprachigkeit“ und E3 „Prosodische Beschränkungen zur phonologischen und morphologischen Entwicklung im bilingualen Spracherwerb“.
- 4 Da die LAPSUS-Benutzerschnittstelle nach heutigen Maßstäben wenig komfortabel und flexibel ist, werden die Daten meist anschließend in modernere DBM-Systeme, d.h. ACCESS oder 4th Dimension, importiert, um deren benutzerfreundlichere Oberflächen nutzen, die Datenstruktur um in LAPSUS nicht vorgesehene Annotationen erweitern oder dort ebenfalls nicht vorgesehene Abfragen durchführen zu können.
- 5 Projekt A3 „Verarbeitung gesprochener Sprache im Dolmetscherprozess“.

tionen (Burger 1997), anzufertigen. Nicht zuletzt ist die Wahl der Transkriptions- und Analysewerkzeuge auch vom Betriebssystem der Rechner des jeweiligen Projektes abhängig – beispielsweise ist syncWriter nur unter Macintosh-Betriebssystemen lauffähig, LAPSUS und HIAT-DOS hingegen nur auf DOS-Betriebssystemen.

Die durchaus wünschenswerte (und auch nicht vermeidbare) Vielfalt der Transkriptionssysteme auf einer konzeptionellen Ebene pflanzt sich also auf der technischen Ebene fort und verursacht dort Probleme, die sich in der praktischen Arbeit des SFB als hinderlich erweisen. Bird/Liberman (2001) sagen zu diesem Thema:

„Particular bodies of data are created with particular needs in mind, using formats and tools tailored to those needs, based on the resources and practices of the community involved. Once created, a linguistic database may subsequently be used for a variety of unforeseen purposes, both inside and outside the community that created it. Adapting existing software for creation, update, indexing, search and display of ‘foreign’ databases typically requires extensive re-engineering.”

Die Vielfalt der benutzten Werkzeuge und Speicherformate, sowie ihre Ausrichtung auf einen bestimmten Analysezweck machen es in den meisten Fällen schwer bis unmöglich, die Daten zweier Projekte auszutauschen, gemeinsam auszuwerten, oder gar mit anderen als den ursprünglich vorgesehenen Werkzeugen zu bearbeiten. Auch wenn Gemeinsamkeiten auf der konzeptionellen Ebene – die trotz aller Unterschiede fast immer vorhanden sind – einen solchen Austausch sinnvoll und gewinnbringend erscheinen lassen würden, stehen seiner Durchführung unüberwindbare technische Hindernisse im Wege. Darüber hinaus besteht in manchen Fällen – genauer: bei der Migration zwischen Betriebssystemen und wenn benutzte Software vom ursprünglichen Anbieter nicht weiterentwickelt wird<sup>6</sup> – die Notwendigkeit, vorhandene Daten in andere Formate zu überführen, alleine um sie weiter nutzbar zu halten.

Ziel des Projektes „Datenbank Mehrsprachigkeit“ ist daher zunächst, eine gemeinsame Basis für alle am Sonderforschungsbereich vorhandenen Transkriptionen gesprochener Sprache zu finden. Die offensichtliche Herangehensweise dafür besteht in der Entwicklung eines geeigneten Datenformates, das im Wesentlichen drei Anforderungen genügen soll:

Es soll *allgemein* genug sein, um verschiedene Konzeptionen von Transkriptionen mit all ihren Unterschieden und Gemeinsamkeiten aufnehmen

6 Dies ist vor allem im Falle des syncWriters ein akutes Problem: Vertrieb und Support des Programms sind Anfang diesen Jahres eingestellt worden. Mit der Einführung des nicht rückwärtskompatiblen MAC OS X-Betriebssystems steht zu befürchten, dass in nicht allzu ferner Zukunft syncWriter-Daten völlig unbrauchbar werden, weil kein Programm mehr zur Verfügung steht, mit dem sie gelesen werden können. Für das dBase III-basierte LAPSUS bestehen ähnliche Probleme.

zu können. Einheiten, die zwei gegebenen Transkriptionssystemen gemein sind, soll es auf die gleiche Art und Weise behandeln und gleichzeitig unterschiedliche Konzepte als solche kenntlich machen. Beispielsweise benutzen die meisten Transkriptionssysteme die Einheit *Wort* in identischer (auf der Orthographie basierenden) Weise, fassen aber mehrere Wörter zu konzeptionell verschiedenen Einheiten (z.B. *Satz*, *Äußerung* oder *Phrasierungseinheit*) zusammen. Die Datenrepräsentation sollte diesen Gemeinsamkeiten und Unterschieden Rechnung tragen.

- Es soll weiterhin *flexibel* an spezielle Analysezwecke anpassbar bleiben. Beispielsweise soll es möglich sein, ein und dasselbe Datum auf unterschiedliche Art und Weise (z.B. in Partitur- und Spaltennotation) graphisch darzustellen oder es mit (möglicherweise einander widersprechenden) verschiedenen Annotationen zu versehen.
- Schließlich soll es dafür sorgen, dass die Daten *langfristig benutzbar* bleiben, d.h. es soll sich offener Standards bedienen und die Daten so weit wie möglich unabhängig von spezieller Software machen.

Das hier vorgestellte System EXMARaLDA (Extensible Markup Language for Discourse Annotation) wurde nach diesen Anforderungen entworfen. Basierend auf der von Bird/Liberman (2001) entwickelten Idee der Annotationsgraphen definiert es mehrere aufeinander aufbauende XML-Formate zu einer inhaltsorientierten Repräsentation von Diskurstranskriptionen verschiedener Komplexitätsstufen. Es dient zunächst als Austauschformat zwischen den vorhandenen projektspezifischen Datenformaten und somit als zentrale Architekturkomponente der mehrsprachigen Datenbank. Gleichzeitig kann es aber, zusammen mit den ebenfalls in diesem Projekt entwickelten Ein- und Ausgabemethoden, auch als eigenständiges, vom Projektzusammenhang losgelöstes Transkriptionssystem<sup>7</sup> betrachtet werden. Abbildung 1 illustriert dies:

<sup>7</sup> Der Terminus „Transkriptionssystem“ wird hier mangels einer besseren Alternative benutzt. EXMARaLDA ist nicht im selben Sinne ein Transkriptionssystem wie HIAT, GAT o.ä. Anders als diese implementiert es nicht ein bestimmtes Diskursmodell, sondern stellt einen Rahmen zur Verfügung, in dem sich verschiedene solcher Transkriptionssysteme auf einer gemeinsamen Basis formulieren lassen.

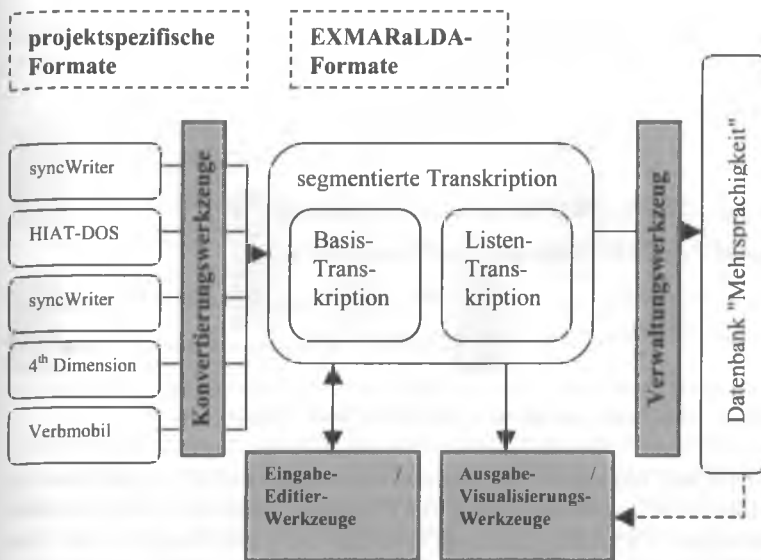


Abbildung 1: *Architektur der Datenbank "Mehrsprachigkeit"*

### 3 EXMARaLDA

Das EXMARaLDA-System kann hier nur in seinen Grundzügen dargestellt werden. Gemäß der texttechnologischen Unterscheidung zwischen logischer und graphischer Struktur geschieht dies in zwei Abschnitten, getrennt nach Datenformaten und Ein- und Ausgabewerkzeugen. Ausführlichere Darstellungen des Systems finden sich in Schmidt (2002a und 2002b).

#### 3.1 Datenformate

##### 3.1.1 Oberflächenstrukturen: basic-transcription und list-transcription

Die meisten herkömmlichen Transkriptionssysteme (also z.B. HIAT und Verbmobil) und die Werkzeuge, mit denen sie umgesetzt werden (syncWriter, HIAT-DOS, Texteditoren) treffen keine strenge Unterscheidung zwischen dem Inhalt und der Darstellung von Transkriptionen. Eine Analyse der von ihnen benutzten graphischen Darstellungsformen stellt daher zunächst den einzigen geeigneten Ausgangspunkt für den Entwurf eines inhaltsorientierten Formates dar. Dabei lassen sich zwei Grundprinzipien unterscheiden.

|          |                             |
|----------|-----------------------------|
| MAX [v]  | Immer unterbrichst Du mich. |
| MAX [nv] | ----- zeigt auf Tom -----   |
| TOM [v]  | Tut mir leid.               |
| TOM [nv] | -- nickt --                 |

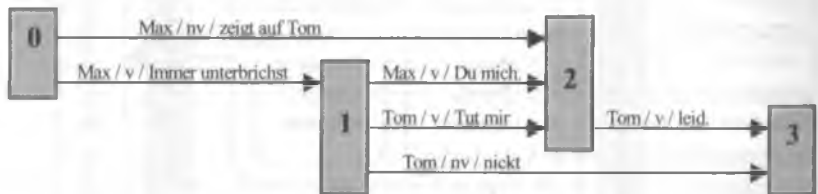


Abbildung 2: Graphische und logische Struktur der basic-transcription

In HIAT und anderen auf der Partiturnotation basierenden Systemen werden Diskursereignisse auf der obersten Ebene in *Spuren* organisiert. Jede Spur erhält im allgemeinen Zuordnung zu einem Sprecher und einer Kategorie und fasst mehrere Beschreibungen von einander nicht überlappenden Diskursereignissen zusammen. Dabei erhält jedes dieser Ereignisse eine von anderen Ereignissen unabhängige Zuordnung zu einer gemeinsamen Zeitachse. Darüber hinaus gehende Struktureinheiten sind für diesen Transkriptionstyp nicht notwendig. Abbildung 2 verdeutlicht dies anhand eines (erfundenen) Transkriptausschnitts.

EXMARaLDA stellt für diesen Transkriptionstyp die *basic-transcription.dtd* zur Verfügung. Eine nach diesen Vorgaben angefertigte Datei würde den obigen Transkriptausschnitt in etwa so kodieren<sup>8</sup>:

```
<timeline>
  <tli id="T0"/>
  <tli id="T1"/>
  <tli id="T2"/>
  <tli id="T3"/>
</timeline>
<tier speaker="MAX" category="v">
  <event start="T0" end="T1">Immer unterbrichst
</event>
  <event start="T1" end="T2">Du mich. </event>
</tier>
<tier speaker="MAX" category="nv">
  <event start="T0" end="T2">zeigt auf Tom</event>
</tier>
[...]
```

<sup>8</sup> Aus Platzgründen wird eine leicht vereinfachte und schematisierte Form der tatsächlichen Syntax verwendet.

**MAX:** (zeigt auf Tom) Immer unterbrichst [Du mich.]  
**TOM:** (nickt) [Tut mir] leid.

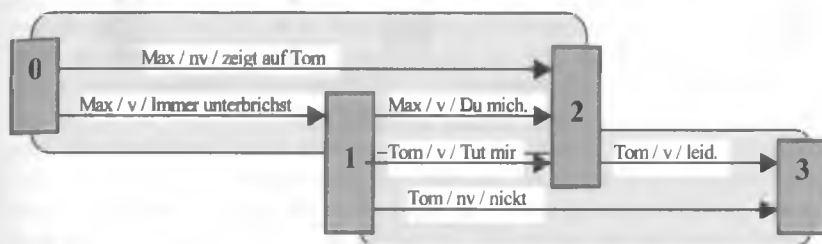


Abbildung 3: Graphische und logische Struktur der list-transcription

Transkriptionssysteme, die mit vertikalen Darstellungsformen arbeiten – also z.B. Verbmobil, aber auch GAT (Selting *et al.* 1998) oder CHAT (MacWhinney 2000) – benutzen hingegen *Sprecherbeiträge* als oberste Struktureinheit. Dabei wird der Diskurs nach sprachlichen Kriterien (z.B. semantische oder syntaktische Vollständigkeit oder intonatorische Kriterien) in Turns, Äußerungen o.ä. unterteilt und andere Ereignisse als von diesen Einheiten abhängig behandelt<sup>9</sup>. Dadurch wird es möglich, eine eindeutige zeitliche Ordnung der obersten Struktureinheiten festzulegen und eine eindimensionale lineare Darstellungsform (im Gegensatz zur zweidimensionalen Darstellungsform der Partiturnotation) zu benutzen. Abbildung 3 illustriert dies.

Für Transkriptionen diesen Typs sieht EXMARaLDA die *list-transcription.dtd* vor. Danach würde der Diskursausschnitt in etwa folgendermaßen kodiert werden:

```
<timeline>
  <tli id="T0"/>
  [...]
</timeline>
<speaker-contribution name="Äußerung" speaker="MAX">
  <main category="v">
    <event start="T0" end="T1">Immer unterbrichst
  </event>
    <event start="T1" end="T2">Du mich. </event>
  </main>
  <dependent category="nv">
    <event start="T0" end="T2">zeigt auf Tom</event>
  </dependent>
</speaker-contribution>
<speaker-contribution name="Äußerung" speaker="TOM">
  <main category="v">
    <event start="T1" end="T2">Tut mir </event>
```

<sup>9</sup> Vgl. z.B. den Terminus „dependent tier“ in MacWhinney (2000).

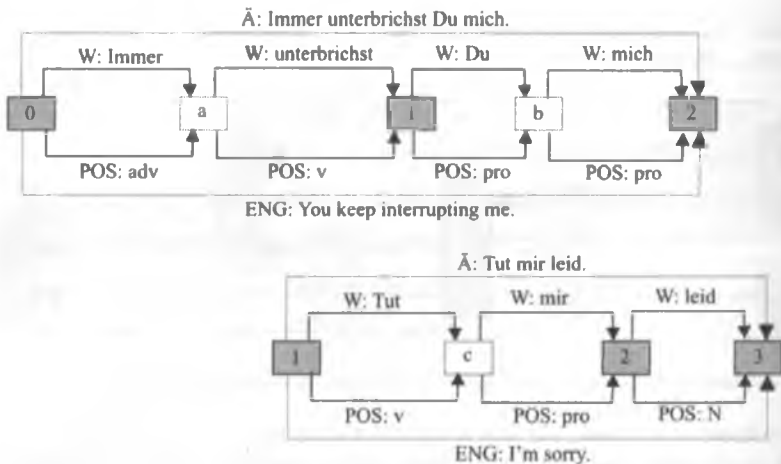


Abbildung 4: Logische Struktur der *segmented-transcription*

```

    <event start="T2" end="T3">leid. </event>
  </main>
  <dependent category="nv">
    <event start="T1" end="T3">nickt</event>
  </dependent>
</speaker-contribution>

```

Wie aus dem Beispiel ersichtlich, enthält eine solche *list-transcription* die von einer *basic-transcription* geforderten Informationen als Teilmenge und ist daher problemlos in eine solche überführbar, also auch in Partiturnotation darstellbar.

### 3.1.2 Zusätzliche Struktur und Annotation: *segmented-transcription*

Die Formate *basic-transcription* und *list-transcription* orientieren sich – wie oben dargestellt – an traditionellen Darstellungstypen für Diskurs-Transkriptionen, wie sie beispielsweise in Edwards (1993) unterschieden werden. Sie sind somit eine ausreichende Grundlage für die Ein- und Ausgabe von Transkriptionen in vertikaler oder Partitur-Notation und werden im EXMARaLDA-System auch in dieser Weise genutzt, d.h. die weiter unten beschriebenen Ein- und Ausgabewerkzeuge arbeiten auf der Basis dieser Formate. Tatsächlich geben sie aber nur einen Teil der Strukturen wieder, die sich dem Betrachter solcher Transkripte erschließen. So mögen Sprecherbeiträge oder Ereignisse sich weiter in Wörter unterteilen oder zusammenfassen lassen, Wörter mögen ihrerseits aus Silben und Phonemen bestehen, und diese Einheiten können alle für die Analyse relevant sein. Insbesondere stellen sie oft den Bezugspunkt von Annotationen, d.h. dem Anreichern der eigentlichen Transkription mit zusätzli-



chen (analytischen) Informationen, dar und müssen daher in irgendeiner Form erschließbar gemacht werden. EXMARaLDA sieht daher als drittes Format die *segmented-transcription* vor, die es erlaubt, auch solche Information zu kodieren. Anders als bei den oben beschriebenen Formaten entstehen so einerseits tiefer geschachtelte Strukturen, andererseits ist eine totale zeitliche Ordnung aller Einheiten oft nicht mehr möglich<sup>10</sup>. Abbildung 4 deutet dies an – Äußerungen werden in Wörter unterteilt (die ihrerseits weitere Unterteilungen in Silben oder Phonemketten haben könnten), und diese Wörter haben untereinander nicht immer eine eindeutige zeitliche Ordnung. So ist zwar eine angemessene Basis für die Annotation der Transkription gewonnen (in der Abbildung wurden Äußerungen mit englischen Übersetzungen, Wörter mit einem Part-of-Speech-Tagging versehen), die wesentlich komplexeren Strukturen machen es aber schwierig, benutzergerechte Ein- und Ausgabemethoden für solche Daten zu konstruieren. EXMARaLDA sieht daher nicht vor, dass Daten direkt in diesem Format eingegeben werden. Vielmehr sollen sie sich automatisch aus einfacheren Strukturen berechnen lassen, beispielsweise durch Segmentierung einer *basic-transcription* mit Hilfe einer regulären Grammatik, die Wort- und Äußerungsgrenzen erkennt. Wie genau solchermaßen segmentierte Transkriptionen sich dann benutzergerecht annotieren und darstellen lassen, bleibt zu erkunden.

### 3.2 Eingabe- und Ausgabewerkzeuge

Auch wenn sich in Form der EXMARaLDA-Formate eine gemeinsame Basis für alle am SFB vorhandenen Transkriptionsdaten gefunden hat, sind die Projekte weiterhin auf Ein- und Ausgabemethoden angewiesen, die ihren spezifischen Bedürfnissen angepasst sind. Die vorhandenen Werkzeuge sind dafür nur bedingt geeignet, teilweise, weil sie schwer weiterzuverarbeitende Daten produzieren (syncWriter, HIAT-DOS), teilweise aber auch, weil sie einfach unkomfortabel in der Bedienung sind (LAPSUS). Es wurden deshalb eigene Ein- und Ausgabemethoden entwickelt, die im Folgenden vorgestellt werden.

#### 3.2.1 Partitureditor

Der Partitureditor präsentiert dem Benutzer eine *basic-transcription* in Form einer editierbaren Endlospartitur, d.h. in mehreren, in ihrer Breite nicht be-

---

<sup>10</sup> Dieser Fall tritt vor allem dann auf, wenn (wie im Beispiel) zwei sich zeitlich überschneidende Sprecherbeiträge jeweils aus mehreren sprachlichen Einheiten bestehen. Beispielsweise mag es zwar theoretisch möglich sein, auch in überlappenden Redebeiträgen einzelne Silben verschiedener Sprecher in eine eindeutige zeitliche Ordnung zu bringen, praktisch kann das aber vom Transkribenten nicht geleistet werden. Die meisten Transkriptionssysteme fordern daher auch nur, Anfang und Ende einer Überlappung zu kennzeichnen und nicht die in ihnen enthaltenen Einheiten zeitlich zu ordnen. Für eine ausführlichere Diskussion dieses Aspekts siehe Schmidt (2002a).

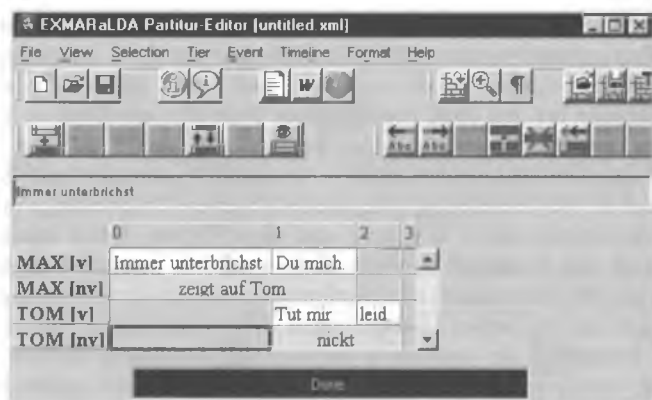


Abbildung 5: Eingabemaske des Partitur-Editors

schränkten Spuren. Die Eingabe des Transkriptionstextes erfolgt in einzelnen Zellen, die gemäß ihrer zeitlichen Abfolge an einer gemeinsamen Zeitachse angeordnet sind. Spuren können bei Bedarf auch nach begonnener Transkription hinzugefügt oder gelöscht werden und lassen sich mit verschiedenen Schriftarten, -größen und -stilen formatieren. Einzelne Ereignisbeschreibungen können zudem mit Bildern oder Ausschnitten aus digitalisierten Ton- oder Video-Aufnahmen verknüpft werden, die bei der Ausgabe in HTML dann als Hyperlinks realisiert werden.

Für die Ausgabe, die wahlweise auf einen Drucker oder in eine RTF- oder eine HTML-Datei erfolgen kann, wird die Endlospartitur auf eine einstellbare Seitenbreite umgebrochen. So exportierte RTF-Dateien lassen sich in gewohnter Weise mit der Word-Textverarbeitung weiterverarbeiten, die HTML-Dateien in allen handelsüblichen Browsern anzeigen.

Seit Version 1.1.1. enthält der Partitur-Editor verschiedene Import-Filter (s. Abschnitt 3.2.3.) für Daten aus anderen Transkriptionseditoren, in einer kommenden Version werden auch Mechanismen zum Überführen der eingegebenen *basic-transcription* in eine *segmented-transcription* in das Programm integriert werden.

### 3.2.2 Simple EXMARaLDA

Simple EXMARaLDA ist eine Eingabemethode, die es erlaubt, einfache Transkriptionen in vertikaler Notation in einen Texteditor einzugeben und anschließend in eine *list-transcription* zu importieren und so für das EXMARaLDA-System nutzbar zu machen. Die Methode besteht lediglich in einer kleinen Anzahl von syntaktischen Vorgaben, die bei der Eingabe beachtet werden müssen, z.B.

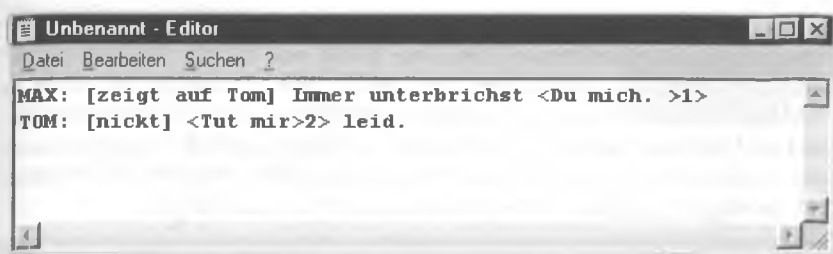


Abbildung 6: Eingabe in Simple EXMARaLDA

dass Sprechersiglen zu Beginn jeder Zeile stehen und mit einem Doppelpunkt abgeschlossen werden, dass Überlappungen mittels spitzer, indizierter Klammern kenntlich gemacht werden, etc. Da sie sich mit jedem Texteditor realisieren lässt, macht diese Eingabemethode den Benutzer weitestgehend unabhängig von spezialisierter Software. Relativ stark gesteuerte Diskurse (z.B. Interviews) lassen sich so mit minimalem Aufwand transkribieren. Für Diskurse mit vielen Überlappungen und wenn non-verbale Kommunikation bezüglich ihrer zeitlichen Ausdehnung detailliert transkribiert werden soll, ist die Methode weniger gut geeignet. Sie kann aber benutzt werden, um schnell eine erste Rohfassung des Transkriptes zu erstellen, die dann in den Partitur-Editor importiert und dort im Detail bearbeitet werden kann.

### 3.2.3 Softwarewerkzeuge dritter

Weder der Partitur-Editor noch die Simple EXMARaLDA-Eingabemethode unterstützen den Transkriptionsvorgang insofern, als sie eine Schnittstelle zur zu transkribierenden Audio- oder Video-Aufnahme zur Verfügung stellen. Es existieren aber eine Reihe von größtenteils ebenfalls noch in der Entwicklung befindlichen Werkzeugen, die genau dies leisten. Der TASX-Annotator (Milde/Gut, in diesem Band), ELAN<sup>11</sup> und PRAAT<sup>12</sup> sind Beispiele für solche Werkzeuge. Auch wenn die von ihnen benutzten Datenformate sich leicht in den Details der Benennungen unterscheiden, folgen sie alle dem gleichen Prinzip, das auch die *basic-transcription* des EXMARaLDA-Systems verwendet. Einzelne Einheiten (*events* bei EXMARaLDA und TASX, *intervals* bei PRAAT, *annotations* bei ELAN) werden über einen Start- und Endpunkt in eine gemeinsame Zeitachse eingeordnet und in übergeordneten Einheiten (*tier* bzw. *layer*) zusammengefasst.

<sup>11</sup> Siehe <http://www.mpi.nl/tools/elan.html>

<sup>12</sup> Siehe <http://www.praat.org>

Aufgrund dieser strukturellen Ähnlichkeit und der guten Zugänglichkeit der verwendeten Speicherformate (XML bei TASX und ELAN, ASCII-Text bei PRAAT) lassen sich Import-Filter, die TASX-, PRAAT- oder ELAN-Daten in EXMARaLDA-Daten (oder umgekehrt) überführen, sehr leicht konstruieren, und die Werkzeuge können so als zusätzliche Eingabemethoden verwendet werden. Import-Filter für TASX- und PRAAT-Daten sind bereits in den Partitur-Editor integriert, weitere werden in kommenden Versionen folgen.

| EXMARaLDA (basic-transcription)  | TASX  |
|--|---|
| <pre>&lt;tier speaker="TOM" cat="v"&gt;   &lt;event start="T1" end="T2"&gt;     Tut mir   &lt;/event&gt;   &lt;event start="T2" end="T3"&gt;     leid.   &lt;/event&gt; &lt;/tier&gt;</pre>  | <pre>&lt;layer l-id="TOM [v]"&gt;   &lt;event start="1.0" end="2.0"&gt;     Tut mir   &lt;/event&gt;   &lt;event start="2.0" end="3.0"&gt;     leid.   &lt;/event&gt; &lt;/tier&gt;</pre>   |
| PRAAT  | ELAN (Eudico Abstract Corpus Model)   |
| <pre>item [3]:   class = "IntervalTier"   name = "TOM [v]"   xmin = 0.0   xmax = 3.0   intervals [1]:     xmin = 1.0     xmax = 2.0     text = "Tut mir "   intervals [2]:     xmin = 2.0     xmax = 3.0     text = "leid. "</pre> | <pre>&lt;tier participant="TOM" [...]&gt;   &lt;annotation&gt;     &lt;[...] timeslot_ref_1="T1"       timeslot_ref_2="T2"&gt;       &lt;value&gt;Tut mir &lt;/value&gt;     &lt;/[...]&gt;   &lt;/annotation&gt;   &lt;annotation&gt;     &lt;[...] timeslot_ref_1="T2"       timeslot_ref_2="T3"&gt;       &lt;value&gt;leid. &lt;/value&gt;     &lt;/[...]&gt;   &lt;/tier&gt;</pre> |

Abbildung 7 gibt noch einmal einen abschließenden Überblick über die verschiedenen Ein- und Ausgabemethoden für EXMARaLDA-Daten<sup>13</sup>.

<sup>13</sup> Die Abkürzung FSM steht für „Finite State Machine“. Die Überführung einer *list-* bzw. *basic-transcription* in eine *segmented-transcription* erfolgt auf der Basis einer Endlichen Maschine, die Äußerungs-, Wortgrenzen etc. erkennt und so, wie oben dargestellt, die Ausgangstranskriptionen mit zusätzlicher Struktur versieht.

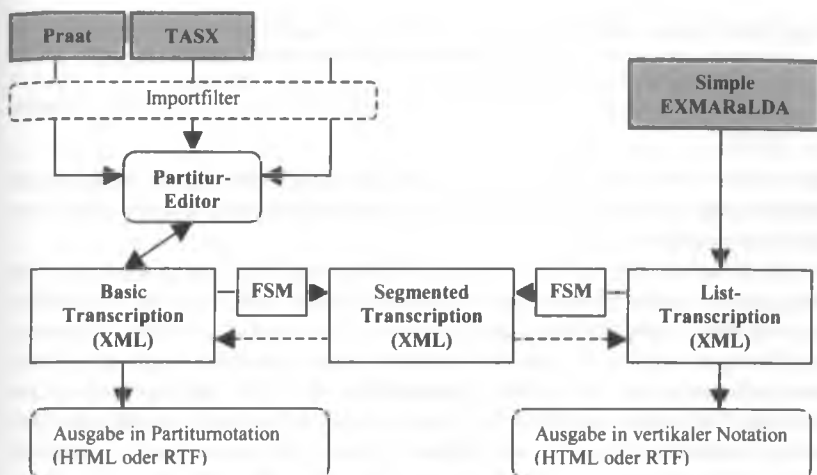


Abbildung 7: Ein- und Ausgabemethoden für EXMARaLDA-Daten

#### 4 Beschreibung von Diskurstranskriptionen mit XML – einige Beobachtungen

Auf den ersten Blick scheint das Feld der computergestützten Diskurstranskription ein Paradebeispiel dafür zu sein, wie der Einsatz von XML zu einer Vereinheitlichung oder gar Standardisierung bei der Kodierung strukturierter Dokumente führen könnte. Wie oben dargestellt hat sich im Laufe der Jahre, bedingt durch unterschiedliche Schwerpunktsetzungen, eine Vielzahl sehr unterschiedlicher Formate herausgebildet, die aber im Grunde alle das Gleiche leisten sollen. Je nachdem, ob ein Diskurs primär als ein Nacheinander von Äußerungen, eine (ungeordnete) Menge von Sätzen oder ein Nebeneinander von Handlungen aufgefasst wird, wurden diejenigen Datenformate gewählt, die diesen Auffassungen konzeptionell am nächsten stehen, also z.B. einfache Textdokumente, Datenbanktabellen oder proprietäre Formate. Die hieraus entstehenden Probleme – Plattform- und Softwareabhängigkeit von Daten, verschiedene Darstellungsformen und Verwendungsweise für ein und dasselbe Datum – wurden oben angedeutet. In einer weit verbreiteten Sichtweise verspricht XML, sie alle auf einen Schlag zu lösen. So heißt es beispielsweise in der Einführung von Ray (2001):

„Mit XML kann so ziemlich jede Art von Information in einer Form gespeichert und organisiert werden, die auf Ihren Bedarf zugeschnitten ist.

Als offener Standard ist XML weder an das Wohl und Wehe eines einzelnen Unternehmens

gebunden, noch ist es mit einer bestimmten Software verknüpft. [...]

Mit seiner klaren, einfachen Syntax und seinen eindeutigen Strukturen ist XML problemlos sowohl von Menschen als auch von Programmen zu lesen und zu analysieren.

XML lässt sich mühelos mit Stylesheets kombinieren, um Dokumente im Format Ihrer Wahl zu erstellen. [...]"

Ich möchte hier anhand der mit EXMARaLDA gewonnenen Erfahrungen andeuten, dass für das Feld der Diskurstranskription (noch) nicht alle diese Versprechen eingelöst werden.

Zunächst ist festzuhalten, dass die Struktur von Diskurstranskriptionen im Vergleich zu vielen anderen Datentypen zweifelsohne sehr komplex ist. Zusätzlich zu den Sequentialitäts- und Inklusionsbeziehungen zwischen Einheiten (Äußerungen oder Wörter können einander folgen, ein Wort ist in einer Äußerung enthalten), die sich auch in geschriebenen Texten finden, müssen in Diskurstranskriptionen auch zeitliche Parallelität (zwei Äußerungen sind simultan) und Äquivalenzbeziehungen (bei Annotationen: z.B. „immer“ ist ein Adverb, die englische Übersetzung der Äußerung „Tut mir leid“ ist „I’m sorry“ etc.) kodiert werden. Alleine mit der XML inhärenten Baumstruktur lassen sich Strukturen dieser Komplexität nicht ausdrücken. Für die Überwindung dieser Beschränkung zeichnen sich zwei grundsätzlich verschiedene Herangehensweisen ab, die sich anhand der Formeln „Dokumente oder Daten?“ und „Bäume oder Graphen?“ gegenüberstellen lassen:

- Eine Herangehensweise besteht darin, Transkriptionen in erster Linie als *Dokumente*, vergleichbar einem „normalen“ geschriebenen Text, anzusehen<sup>14</sup>. Wie einem Text kann ihnen eine hierarchische Struktur zugewiesen werden, und die Rolle von XML besteht vor allem darin, diese Struktur dem eigentlichen Text in Form eines Markup-Baums *hinzuzufügen*. Eine solches Dokument erfüllt tatsächlich die meisten der oben aufgeführten Versprechungen – es ist, zumindest bis zu einem gewissen Grad, für Menschen lesbar, lässt sich mit Stylesheets relativ mühelos in verschiedene Präsentationsformate überführen und kann im Prinzip ohne *spezialisierte* Software, also beispielsweise in einem Texteditor oder einem generischen XML-Editor, erstellt werden. Allerdings vernachlässigt diese Herangehensweise zunächst parallele Strukturen. Weil diese sich nicht direkt mit den XML inhärenten Mitteln kodieren lassen, müssen Behelfsmechanismen gefunden werden, die dann ihrerseits interpretationsabhängig sind. Ob diese

<sup>14</sup> Beispiele für diese Herangehensweise sind die TEI-Richtlinien zur Kodierung von Transkriptionen gesprochener Sprache (Sperberg-McQueen/Burnard 1999) oder das von der LTG in Edinburgh vorgeschlagene Konzept der *Standoff-Annotation* (Carletta/Isard/McKelvie 2000). Die oben beschriebene EXMARaLDA *list-transcription* weist ebenfalls in diese Richtung gehende Züge auf.

Form der Datenrepräsentation dann wirklich noch „auf den Bedarf zugeschnitten ist“, lässt sich also bestreiten.

Die entgegengesetzte Herangehensweise manifestiert sich im Konzept der Annotationsgraphen. Hier wird die Auffassung vertreten, dass Diskurstranskriptionen in erster Linie gerade *keine* Baumstruktur, sondern die Struktur gerichteter, azyklischer *Graphen* aufweisen. Wie Bird/Liberman (2001) zeigen, sind diese Graphen geeignet, alle für Diskurstranskriptionen charakteristischen Beziehungen in vereinheitlichter Art und Weise zu formulieren. Diese vereinheitlichte Struktur lässt sich allerdings nicht unmittelbar auf einen XML-Baum abbilden. Diskurstranskriptionen, die gemäß dem Prinzip der Annotationsgraphen in XML kodiert sind<sup>15</sup>, sind daher weder für den Menschen gut lesbar (denn die von XML erzwungene Baumstruktur spiegelt nicht die eigentliche Struktur der Dokumente wieder), noch lassen sie sich über Stylesheets einfach in Präsentationsformate überführen (denn Stylesheets sind in erster Linie dazu geeignet, Baumstrukturen in andere Baumstrukturen zu überführen). XML wird hier also lediglich in seiner Eigenschaft als plattformunabhängiges Speicherformat genutzt, was dem vielfach thematisierten „*daten-zentrischen*“ Verständnis (im Gegensatz zum „*dokument-zentrischen*“, vgl. z.B. Box *et al.* 2000) entspricht.

Es ist also offensichtlich, dass die im Zusammenhang mit XML oft gestellte Frage „Dokumente oder Daten?“ sich für das Feld der Diskurstranskription nicht eindeutig beantworten lässt – sie lässt sich höchstens auf die spezifizierte Fassung „Bäume oder Graphen?“ eingrenzen. Interessanterweise lassen sich hier Parallelen zu einer wesentlich älteren Kontroverse, der in der Literatur viel diskutierten Frage „Zeilen- oder Partiturnotation?“, ziehen. Die Zeilennotation steht dem geschriebenen Text näher und ist somit – unabhängig von einer computertechnischen Umsetzung – leichter mit herkömmlichen Vorstellungen von Text zu vereinbaren als die Partiturnotation. Diese wiederum erhebt für sich den Anspruch, gerade diejenigen Strukturen, die gesprochene von geschriebener Sprache unterscheiden, angemessener repräsentieren zu können. Wie die oben beschriebenen Formate *basic-transcription* und *list-transcription* erkennen lassen, muss sich das EXMARaLDA-System mit beiden Sichtweisen auseinandersetzen und würde daher von texttechnologischen Antworten auf die hier gestellten Fragen sicherlich profitieren.

## 5 Literatur

Bird, Steven und Mark Liberman: A formal framework for linguistic annotation. In: Speech Communication 33 (1, 2), 2001, S. 23-60.

<sup>15</sup> Also z.B. Dokumente im AIF-Format (Bird/Liberman 2001), die EXMARaLDA *basic-transcription* und die in Abschnitt 3.2.3. angesprochenen TASX-, ELAN und Praat-Formate.

- Bird, Steven, Peter Buneman und Mark Liberman* (Hg.): Proceedings of the IRCS Workshop On Linguistic Databases, 11-13 December 2001, Institute for Research in Cognitive Science, University of Pennsylvania, Philadelphia 2001.
- Box, Don, Aaron Skonnard und John Lam*: Essential XML: Beyond Markup. Boston: Addison-Wesley 2000.
- Burger, Susanne*: Transliteration spontansprachlicher Daten. Verbomobil Technisches Dokument 56. München 1997.
- Carletta, Jean, Amy Isard und David McKelvie*: Linguistic Data Processing For Everyman. Paper presented at the workshop on Web-Based Language Documentation and Description, Philadelphia 2000.
- Crysmann, Berthold*: LAPSUS: A utility for the transcription of empirical data in language acquisition research. Hamburg 1995. [Unveröffentlichtes Manuskript.]
- Edwards, Jane und Martin Lampert* (Hg.): Talking Data – Transcription and Coding in Discourse Research. Hillsdale 1992.
- Edwards, Jane*: Principles and Contrasting Systems of Discourse Transcription. In: *Edwards, Jane und Martin Lampert* (Hg., 1992), S. 3-31.
- Elhlich, Konrad*: HIAT - a Transcription System for Discourse Data. In: *Edwards, Jane und Martin Lampert* (Hg., 1992), S. 123-148.
- Elhlich, Konrad und Jochen Rehbein*: Halbinterpretative Arbeitstranskriptionen (HIAT). In: *Linguistische Berichte* 45, 1976, S. 21-41.
- MacWhinney, Brian*: The CHILDES project – tools for analysing talk. Mahwah, NJ u.a.: Lawrence Erlbaum 2000.
- Milde, Jan-Torsten und Ulrike Gut*: The TASX-Environment: an XML-based corpus database for time aligned language data. In: *Bird, Steven, Peter Buneman und Mark Liberman* (Hg.): Proceedings of the IRCS Workshop On Linguistic Databases, 11-13 December 2001, Institute for Research in Cognitive Science, University of Pennsylvania, Philadelphia 2001, S. 174-180.
- Ray, Erik*: Einführung in XML. Köln: O'Reilly 2001.
- Rehbein, Jochen, Wilhelm Griefhaber, Petra Löning, Marion Hartung und Kristin Bührig*: Manual für das computergestützte Transkribieren mit dem Programm SyncWRITER nach dem Verfahren der Halbinterpretativen Arbeitstranskriptionen (HIAT). Hamburg: Germanisches Seminar der Universität Hamburg 1993.
- Schmidt, Thomas*: The transcription system EXMARaLDA: An application of the annotation graph formalism as the Basis of a Database of Multilingual Spoken Discourse. In: *Bird, Steven, Peter Buneman und Mark Liberman* (Hg., 2001), S. 219-227.
- Schmidt, Thomas*: EXMARaLDA – ein System zur Diskurstanskription auf dem Computer. In: *Arbeiten zur Mehrsprachigkeit (AZM)*, Serie B. Hamburg 2002a.
- Schmidt, Thomas*: Gesprächstranskription auf dem Computer – das System EXMARaLDA. In: *Gesprächsforschung – Online-Zeitschrift zur verbalen Interaktion* 3. Freiburg 2002b, S. 1-23.
- Selling, Margret, Peter Auer, Birgit Barden, Jörg Bergmann, Elizabeth Couper-Kuhlen, Susanne Günthner, Christoph Meier, Uta Quasthoff, Peter Schlobinski und Susanne Uhmann*: Gesprächsanalytisches Transkriptionssystem (GAT). In: *Linguistische Berichte* 173, 1998, S. 91-122.
- Sperberg-McQueen, C.M. und Lou Burnard* (Hg.): Guidelines for Electronic Text Encoding and Interchange. Oxford 1999.